IN THE SPECIFICATION:

Please delete the information beginning on page 1, line 2 as follows:

~~Inventors: Greg Lindhorst~~
~~John Buchler~~
~~Steve Millet~~
~~John Shewchuck~~
~~David Johnson~~

Please replace the paragraph beginning on page 1, line 4 with the following:

This application is a Continuation of U.S. Application Serial No. 09/223,565 filed on December 31, 2998, hereby incorporated by reference as to its entirety. The present application is related to U.S. Serial No. 08/959,300, entitled "Method and Apparatus for Automatic Generation of Text and Computer-Executable Code", filed on October 28, 1997. Also, the present application is related to U.S. Serial No. 09/223,773, "A System for Converting Event-Driven Code into Serially Executed Code", filed on December 31, 1998; to U.S. Serial No. 09/223,558, entitled "Transporting Objects Between A Client and A Server', filed on December 31, 1998; and, to U.S. Serial No. 09/223,774, entitled "Page Object Model", filed on December 31, 1998.

Please replace the paragraph beginning on page 3, line 4 with the following:

ASPs may be called using a URL (uniform resource locator) which may refer to the ASP file "page.asp" as follows: "<http: //www. anycompany. com/ page.asp>." While the following description uses the term "active server page" or "ASP," any document or page which contains a script which, when executed, pulls data from a database and forwards it to an end application (another document or page) falls within the scope of the term, ASP, as used herein.

Please replace the paragraph beginning on page 5, line 7 with the following:

There is currently no way to incorporate the functionality of a destination ASP into a current page. Placing the following tag on a client's page "<A HREF = "http: //www. anycompany. com/ alpha/page.asp">" allows the user to select the tag and navigate to the ASP names "page.asp" in the "alpha" directory of the www.anycompany.com site. However, once the page.asp is selected with specific parameters, the current page is lost with all its information. The only way to return to the current page and modify the information transmitted to the ASP is by backtracking and re-entering the previously entered parameters. While object-oriented programming exists with respect to other programming areas, there is little or no support for objects in program files of HTML pages including ASPs.

Please replace the paragraph beginning on page 20, line 1 with the following:

Figure 5 shows composition space 1002 and interaction space 1003 as including a non-volatile memory1030 in at least one of the two spaces 1002 and 1003. An ASP page 1045 containing objects is stored in non-volatile memory 1030. The ASP page 1045 includes objects balloon, cube, and slab. These objects may have associated methods and properties. In response to a request from a browser in interaction space, the ASP page 1045 executes and instantiates (step 1031) the object "balloon" with stored property information from non-volatile memory 1030 so as to create balloon$_1$ 1032. Balloon$_1$ 1032 is then transmitted (step 1033) to interaction space 1003 and appears as balloon$_1$ 1034 with persistence information 1035P. In this example, the object balloon$_1$ 1034 is transmitted with its own code to interaction space 1003. The persistence information 1035P that accompanies balloon$_1$ 1034 describes the properties of received balloon$_1$ 1034. Here, the properties are (object number =1, color = red, radius =5).

Next, in the interaction space 1003, the browser receives an event 1036 (for example, a request to paint the balloon_1 1034 blue). The balloon_1 1034 has not changed but, having received event 1036, is identified as balloon_1 1035. The browser transmits (step 1038) the event 1036 to the composition space to handle the event. The information transmitted to the composition space 1002 include the event 1036 (paint blue) and the persistence information 1035P which identifies the balloon_1 1035 to the ASP in composition space 1002. Next, the ASP re-instantiates (step 1039) the object balloon_1 as 1041 and handles the event 1036 (paint blue) in step 1040. The object balloon_1 is transformed into balloon_2 1042. The ASP then transmits balloon_2 1042 (in step 1043) back to the interaction space 1003 with persistence information (object =1, color = blue, radius =5) 1045P. Finally, the balloon_2 is received as balloon_2 1044 and displayed to user. In this example, it is noted that at least one advantage of storing the persistence information in the interaction space 1003 is that the interaction space is the best place to maintain this information.

Please replace the paragraph beginning on page 28, line 15 with the following:

While editing "apple" in step 301, the developer references pageb.asp (step 302). Noting that pageb.asp does not exist, the editor creates pageb.asp in step 303. Created pageb.asp is shown as page 310. After step 303, the editor inserts a Page Object DTC 311 into pageb.asp 310 in step 304. Next, if possible, the editor can attempt to populate the fields of the various property pages of new pageb.asp. Otherwise, the editor prompts the developer to describe the various methods and properties (and other information) relevant to page (step 305). The editor may then prompt the developer for objects 312 (with methods eat, sleep, fly) to be included on page 310 (step 306). For the inclusion of objects, the developer may drag-and-drop in old objects to create a new collection of objects on the page 310. As indicated by dotted lines, next step may be included to describe the methods and properties of each new object 312 on the page 310 (step 307). The developer then proceeds to save the modifications made to the various pages (pagea.asp, pageb.asp) (step 308). The editor processes the DTCs to create appropriate scripts in new client and server pages and ASPs (step 309).

Please replace the paragraph beginning on page 30, line 7 with the following:

Figure 9 relates to the execution of a method with no new navigation in the client's browser. Figure 9 shows the receipt of a page in step 501. Next, the end user selects a method with the name of <foo.execute.method ()> in step 502. In contrast to step 403, the browser keeps the current page (step 503), and grabs the new page (504), which accesses the method *method.* Finally, while the method *method* may be running elsewhere (or it may be running on the client), the client's browser receives the data generated by the method and returns them to the currently running page (505). To the extent that information is displayed by the new method, the method effectively appears to run the current page.

Please replace the paragraph beginning on page 35, line 18 with the following:

Figure 11 shows an ASP proxy object 704 located on server 701. When an ASP object is loaded by a server, the property values are retrieved 706 (from wherever they are stored 703) and are placed into the proxy objects properties 704. Also, the proxy object's properties for the client are prepared and stored in ASP object 704 as well. During construction of the page, the proxy objects' properties may be accessed and modified by the server script. If modified, the properties may be sent back 707 to the data storage 703 for future retrieval.